
Added SRAM performances benefits for STM32F413/F423 lines

Introduction

In embedded designs, the need for large quantities of data increases. Memory, especially RAM has become a very precious resource, and understanding the best use of this memory is crucial to achieving optimal performance.

This application note aims to help users to benefit from the increased SRAM performance and lower power consumption of the STM32F413/F423 microcontrollers by studying different scenarios.

The main goal is to explain the architectural design and system module features that can be tuned to optimize an application.

Contents

- 1 System architecture overview for STM32F413/F423 5**
 - 1.1 STM32F413/F423 core buses 5
 - 1.2 Embedded SRAM 6
 - 1.3 Comparative study of performance and power consumption 6
 - 1.3.1 First scenario: Code executed from Flash and data stored in SRAM1 . . 7
 - 1.3.2 Second scenario: Code executed from SRAM2 and data stored in SRAM1 8
 - 1.3.3 Comparative study of performance results 9
 - 1.3.4 Comparative study of power consumption results 9

- 2 Conclusion 10**

- 3 Revision history 11**



List of tables

Table 1.	Performance results summary	9
Table 2.	Power consumption results summary	9
Table 3.	Document revision history	11

List of figures

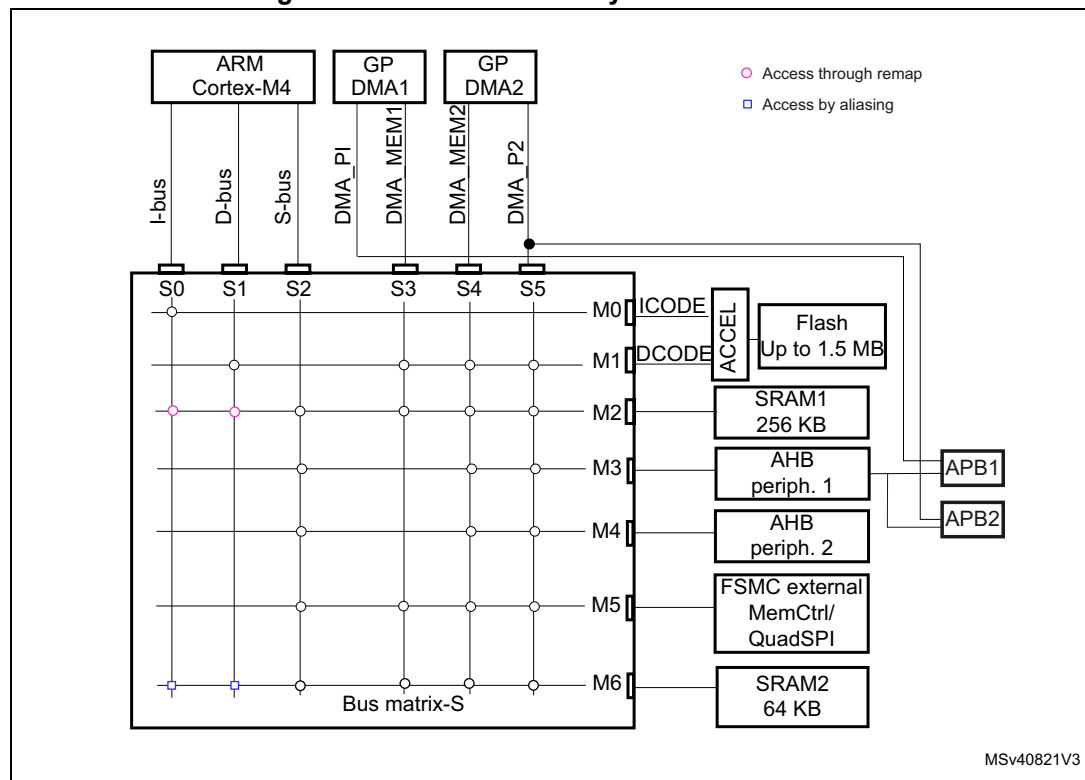
Figure 1.	STM32F413/F423 system architecture	5
Figure 2.	Scenario 1: Performance configuration	7
Figure 3.	Scenario 1: MDK-ARM scatter file	7
Figure 4.	Scenario 2: Performance configuration	8
Figure 5.	Scenario 2: MDK-ARM scatter file	8

1 System architecture overview for STM32F413/F423

The first step to optimize the system performance is to understand the high level architecture.

Figure 1 shows a simplified block diagram of the STM32F4 Series.

Figure 1. STM32F413/F423 system architecture



1.1 STM32F413/F423 core buses

The STM32F413/F423 devices are based on Cortex[®]-M4 core with FPU (Floating point unit) core, which uses a Harvard architecture with some memory mapped buses.

The bus matrix provides access from a master to a slave, enabling concurrence access and efficient operation.

- I-bus: Instruction bus used by the core to fetch instructions from a memory containing code (internal Flash memory / SRAM).
- D-bus Data bus used by the core for literal load and debug access to/from a memory containing code or data (internal Flash memory / SRAM).
- S-Bus System bus used by the core to access data located in a peripheral or in SRAM. Instructions may also be fetch on this bus. The targets of this bus are internal SRAM, the AHB 1 peripherals including the APB peripherals and the AHB2 peripherals.

1.2 Embedded SRAM

The STM32F413/F423 feature a 320 K bytes of SRAM.

The embedded SRAM is divided into two blocks:

- SRAM1 mapped at address 0x2000_0000 and accessible by all AHB masters.
- SRAM2 mapped at address 0x2004_0000 and accessible by all AHB masters.

In the STM32F413/F423, SRAM2 can be accessed as following:

- Instruction fetches can be performed over the I-Code bus. Data accesses are performed over the D-Code bus.
- Instruction fetches and data accesses can be performed over the system bus.
- Data accesses are aliases. Instruction accesses are not aliases.

For the execution of load or store instructions, separate buses are helpful because instruction fetch and data access are performed at the same time, and no Bus matrix arbitration is inserted.

CPU can access SRAM2 memory via I-bus and D-bus, when SRAM2 is mapped at the address range: 0x1000 0000 to 0x1000 FFFF.

The System bus cycle timing depends on the type of access:

- The system bus data accesses have no delay added at the core.
- The instruction accesses add one wait state at the core.

Placing code and data into both SRAM blocks helps to increase the parallelism and the overall performance.

Note: For a typical application, the best performance will grant by inserting the critical code in SRAM2 and the data and the stack in SRAM1.

1.3 Comparative study of performance and power consumption

The comparative studies of performance and power consumption is done with the Core-Mark algorithm.

This algorithm is performed following the above suggestions and the below system configurations:

System configurations:

1. Execution from Flash
2. Execution from SRAM

Test Conditions:

- The tests are performed at ambient temperature and $V_{DD} = 3.3V$.
- In case of funning from Flash,
 - ART is enabled
 - 3 wait state (WS) are added.

1.3.1 First scenario: Code executed from Flash and data stored in SRAM1

Figure 2 shows the first scenario:

Figure 2. Scenario 1: Performance configuration

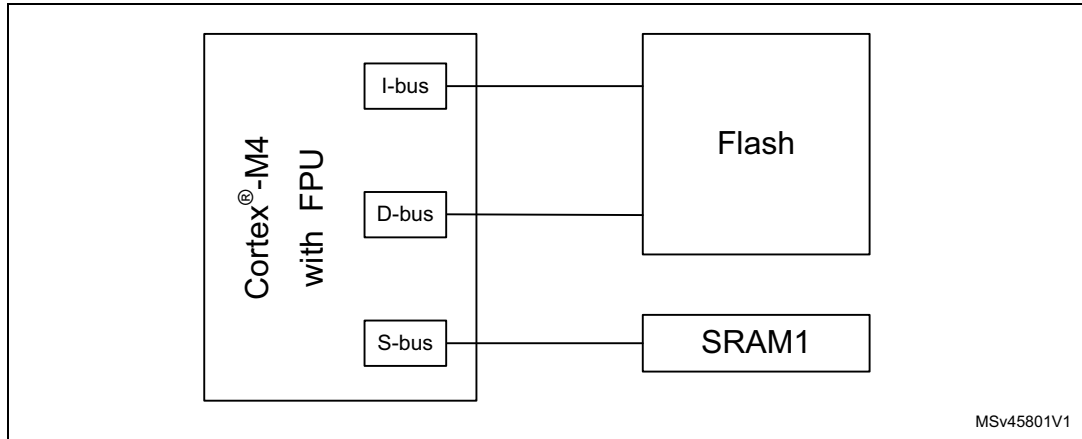


Figure 3 shows the MDK-ARM scatter file.

Figure 3. Scenario 1: MDK-ARM scatter file

```
1 ; *****
2 ; *** Scatter-Loading Description File generated by uVision ***
3 ; *****
4
5 LR_IROM1 0x08000000 0x00180000 { ; load region size_region
6   ER_IROM1 0x08000000 0x00180000 { ; load address = execution address
7     *.o (RESET, +First)
8     *(InRoot$$Sections)
9     .ANY (+RO)
10  }
11  RW_IRAM1 0x20000000 0x00050000 { ; RW data
12    .ANY (+RW +ZI)
13  }
14 }
```

1.3.2 Second scenario: Code executed from SRAM2 and data stored in SRAM1

Figure 2 shows the second scenario:

Figure 4. Scenario 2: Performance configuration

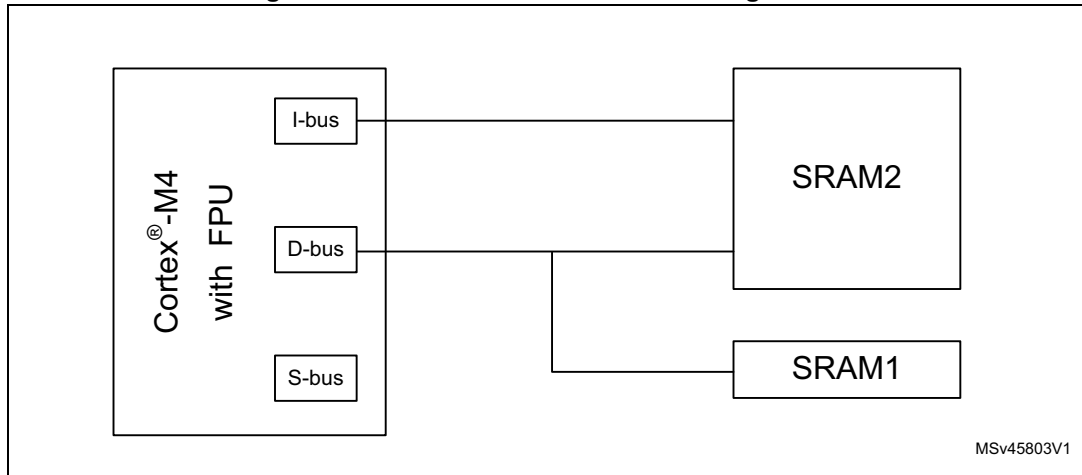


Figure 5 shows the MDK-ARM scatter file obtained by KEIL MDK.

Figure 5. Scenario 2: MDK-ARM scatter file

```
1 ; *****
2 ; *** Scatter-Loading Description File generated by uVision ***
3 ; *****
4
5 LR_IROM1 0x10000000 0x8000 { ; load region size_region
6 ER_IROM1 0x10000000 0x8000 { ; load address = execution address
7 *.o (RESET, +First)
8 *(InRoot$$Sections)
9 .ANY (+RO)
10 }
11
12 RW_IRAM1 0x20008000 0x4000 { ; RW data
13 .ANY (+RW +ZI)
14 }
15 }
16 }
17
```


1.3.3 Comparative study of performance results

Table 1 shows the results obtained:

Table 1. Performance results summary

-	System clock 100 MHz	
	Code executed from FLASH	Code executed from SRAM2 and DATA stored in SRAM1
Score	3.39	3.41

1.3.4 Comparative study of power consumption results

Table 2 shows the results obtained:

Table 2. Power consumption results summary

-	System clock 100 MHz	
	Code executed from FLASH	Code executed from SRAM2 and DATA stored in SRAM1
Power Consumption [mA]	22.91	21.54

Note: The best performance and the lowest power consumption are obtained in the second configuration: code executed from SRAM2 and data stored in SRAM1.

2 Conclusion

This application note complements the STM32F413/F423 datasheets and reference manual by describing techniques to optimize performance and power consumption with added SRAMs.

3 Revision history

Table 3. Document revision history

Date	Revision	Changes
18-Apr-2017	1	Initial release.

IMPORTANT NOTICE – PLEASE READ CAREFULLY

STMicroelectronics NV and its subsidiaries ("ST") reserve the right to make changes, corrections, enhancements, modifications, and improvements to ST products and/or to this document at any time without notice. Purchasers should obtain the latest relevant information on ST products before placing orders. ST products are sold pursuant to ST's terms and conditions of sale in place at the time of order acknowledgement.

Purchasers are solely responsible for the choice, selection, and use of ST products and ST assumes no liability for application assistance or the design of Purchasers' products.

No license, express or implied, to any intellectual property right is granted by ST herein.

Resale of ST products with provisions different from the information set forth herein shall void any warranty granted by ST for such product.

ST and the ST logo are trademarks of ST. All other product or service names are the property of their respective owners.

Information in this document supersedes and replaces information previously supplied in any prior versions of this document.

© 2017 STMicroelectronics – All rights reserved